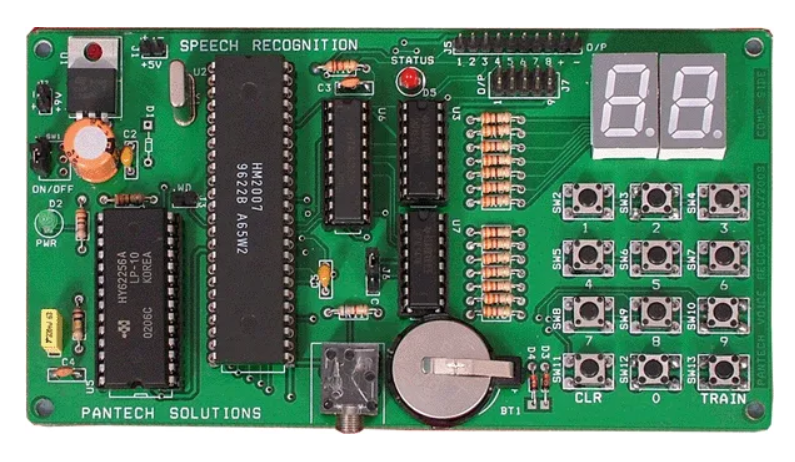
**Circuit Diagram:**



[Speech Recognition](https://www.geeksforgeeks.org/automatic-speech-recognition-using-whisper/) is the capacity of a machine or program to recognize spoken words and transform them into text. [Speech Recognition](https://www.geeksforgeeks.org/automatic-speech-recognition-using-whisper/) is an important feature in several applications used such as home automation, artificial intelligence, etc. In this article, we are going to explore how speech recognition software work, speech recognition algorithms, and the role of NLP.

**Code:**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <pocketsphinx.h>

#include <sphinxbase/ad.h>

#include <sphinxbase/err.h>

ps\_decoder\_t\* init\_decoder() {

ps\_decoder\_t \*ps;

cmd\_ln\_t \*config;

const char \*acmod = "/path/to/acoustic/model

const char \*dict = "/path/to/dictionary";

const char \*lm = "/path/to/language/model";

config = cmd\_ln\_init(NULL, ps\_args(), TRUE,

"-hmm", acmod

"-dict", dict,

"-lm", lm,

NULL);

if (config == NULL) {

fprintf(stderr, "Error initializing configuration\n");

return NULL;

}

ps = ps\_init(config);

if (ps == NULL) {

fprintf(stderr, "Failed to create recognizer\n");

return NULL;

}

return ps;

}

int main() {

ps\_decoder\_t \*ps = init\_decoder();

if (ps == NULL) {

return 1;

}

ad\_rec\_t \*ad = ad\_open\_dev(NULL, 16000

if (ad == NULL) {

fprintf(stderr, "Failed to open audio device\n");

ps\_free(ps);

return 1;

}

if (ad\_start\_rec(ad) < 0) {

fprintf(stderr, "Failed to start recording\n");

ad\_close(ad);

ps\_free(ps);

return 1;

}

printf("Listening... Speak into the microphone.\n");

int16 adbuf[2048];

int32 score;

char const \*hyp;

while (1) {

int len = ad\_read(ad, adbuf, sizeof(adbuf) / sizeof(adbuf[0]));

if (len < 0) {

fprintf(stderr, "Error reading audio\n");

break;

}

ps\_process\_raw(ps, adbuf, len, FALSE, FALSE)

hyp = ps\_get\_hyp(ps, &score);

if (hyp != NULL) {

printf("Recognized: %s\n", hyp);

}

}

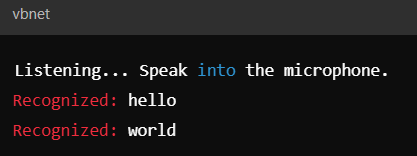
ad\_close(ad);

ps\_free(ps);

return 0;

}

**Output Demonstration:**

****